Available online at www.sciencedirect.com

SCIENCE DIRECT®

Electronic Notes in Theoretical Computer Science

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Preface

Volume 10

Gordon Andrew, Pitts Carolyn and Talcott Andrew

Abstract

This issue of ENTCS is an unrefereed conference record of talks presented at the Second Workshop on Higher Order Operational Techniques in Semantics (HOOTS II) held at Stanford University, December 8-11, 1997. The meeting was organised by A. Gordon, A. Pitts and C. Talcott with generous sponsorship from Harlequin Ltd, NSF and ONR. The first HOOTS workshop was held October 28-30, 1995 as part of the University of Cambridge Isaac Newton Institute research programme on Semantics of Computation (July-Dec 1995).

The study of operational techniques for higher-order languages is now a thriving area, with much research activity going on world-wide. An important open problem is a theory of program equivalence for languages with higher-order features, including functions and objects. Techniques for defining and reasoning about equivalence and other properties of higher-order programs have emerged in distinct communities, including the concurrency, functional programming and type theory communities. The purpose of the HOOTS workshops was to bring researchers from these communities together to discuss current trends in the theory of operational semantics, its application to higher-order languages and its connection with more established semantic techniques.

Papers presented at HOOTS II covered a broad range of topics:

- techniques such as bisimulation and logical relations for reasoning about contextual equivalence
- alternative program relations such as operational subsumption, and evaluation rules for program contexts
- operational models including adaptation of big-step evaluation semantics to provide capabilities of small-step and denotational semantics forms, flow graphs, and history dependent automata
- higher-order programming calculi including: imperative call-by-need lambda calculus, action calculi, process calculi for reasoning about mobility and security, interaction of actors and pi calculus agents
- approaches to program analysis and verification, including: logics for control flow analysis, monadic type systems, and diagramatic specification notation for actor systems;
- programming environment tools such a type systems for Java byte-code, and higher-order program units for modularity.

Programs and participants lists for HOOTS I and II and other information about HOOTS, past and future can be found here

1571-0661
10.1016/S1571-0661(05)80684-X